JAMES COOK
UNIVERSITY
AUSTRALIA

CP1406 Week 4 Practical – HTML and CSS

To improve the appearance of a website by including colour, formatting text, and adding margins, borders, and shadows, for example, you need to apply styles created with Cascading Style Sheets (CSS), a language you use to describe the formatting of a document written in a markup language such as HTML. While HTML provides the structure and content of a webpage, you use CSS styles to determine the formatting for a webpage. Today you will apply styles to the HTML elements on webpages to give the site a certain look and feel.

For today's practical, you need to submit:
- **A single zip file** containing
  - **The Fitness folder**. The "Fitness" folder includes:
    - HTML files (index.html, template.html, contact.html, and aboutus.html) you have created for the site
    - a CSS file for the "Fitness" website
    - the "images" folder contains images for the "Fitness" website
  - **The Training Zone folder**. The "Training Zone" folder includes:
    - HTML files you have created for the site
    - a CSS file for the "Training Zone" website
    - the "images" folder contains images for the "Training Zone" website

Part 1 – Adding Comments to a CSS File

Recall that comments are used to explain the code, and may help when you edit the source code later. A CSS comment starts with **/*** and ends with ***/**.

When you create a CSS file, including comments provides additional information to help you understand your style rules. You can also use a comment to identify that you are the author of the style sheet.

- Open your editor, create, and name this new file "**styles.css**"; ensure that you save it as **Cascading Style Sheets** or **CSS** inside the folder called "**css**" which is inside the "Fitness" folder.
- On the first line of the file, add the following code (Figure 1):

```
/*
Author:  your first and last name
Date:   the current date
File Name: styles.css
*/
```

You will see many comments in this prac. Unless you are explicitly asked to create comments, you can leave out these CSS comments in your files.

```
1 ▼ /*
2    Author: Student's Name
3    Date: Current Date
4    File Name: styles.css
5    */
```

Figure 1

Part 2 – Creating a Style Rule for the Body Element

To apply styles to the Forward Fitness Club website, you can create them in an **external style sheet**. The styles defined in the external style sheet will apply to all the webpages in the website.

- In styles.css, add the following code to create a style rule for the body element (Figure 2):

```
body {  /* Add the body selector */
        background-color: #ccc;  /* Set the background colour */
} /* Closing brace for the body selector */
```

Note: If your editor did NOT provide autocomplete suggestions as you typed your declaration (e.g., as you started typing "backgro"… it showed you all the properties that start with "backgro"… then you should change to a better editor, like **Visual Studio Code**, which makes this much easier because it knows all of these and you don't have to type them. Not typing full property and value names means you can mostly avoid spelling mistaks ☺.
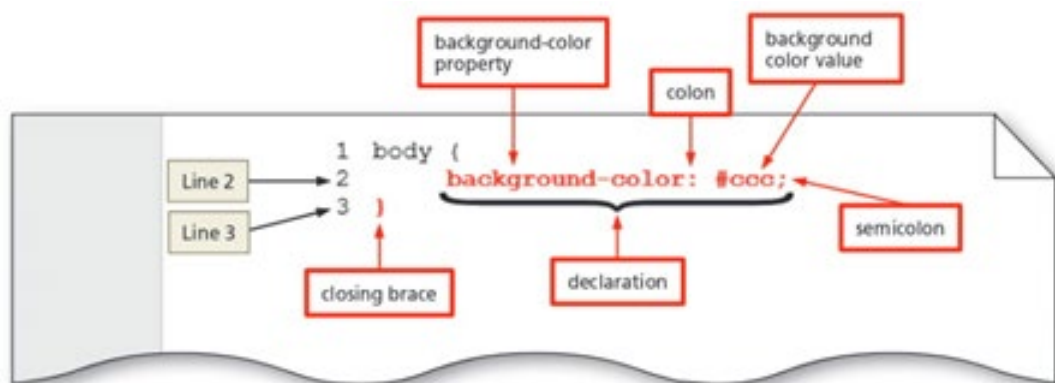
- Save your changes.

Figure 2

Part 3 – Linking HTML Pages to the CSS File

To apply the CSS style rules to a webpage, **you must link the webpage to the CSS file**. To view the applied styles, open the home page (**index.html**) in a browser. To link the **styles.css** file to an HTML page, you include a link to the style sheet within the **head** section of the document.

- Open **index.html** in your editor to prepare to link the page to the **CSS file**.
- Before the closing </head> tag, add the following code to create a link to the style sheet file (Figure 3):
- (Note: you may or may not have put your styles in the "css" folder. If you have used the subfolder, include it in the href, otherwise don't include the folder name.)

<link rel="stylesheet" href="styles.css">

```
1 <!DOCTYPE html>

2 <!-- This website template was created by: Student's First
    Name Student's Last Name -->

3 <html lang="en">

4 <head>

5     <title>Forward Fitness Club</title>

6     <meta charset="utf-8">

Line 7  → 7     <link rel="stylesheet" href="css/styles.css">   ← link to styles.css

8 </head>
```

Figure 3

- Reload your web browser to see the effect of this. If you see no change, go back and see what might be missing or incorrect and try and fix it.
  Ask your neighbour for help if you can't solve the problem.

Part 4 – Positioning Elements

Return to your CSS file to create style rules for the **header**, **nav**, and **main** elements.

- In the text editor, return to the **styles.css** file.
- Below the closing brace of the body selector, add the following code (Figure 4):

```
header {  /* add the header selector */
        width: 25%;  /* Set the width of the element */
        float: left;  /* Add a declaration that sets a left float */
} /* Closing brace for the header selector */

nav {
        width: 70%;
```

```
        float: right;
        margin: 5em 3em 0 0; /* Set the margin on all sides of the element:
top, right, bottom, left */
}

main {
        clear: left;  /* clears a float to the left */
}
```
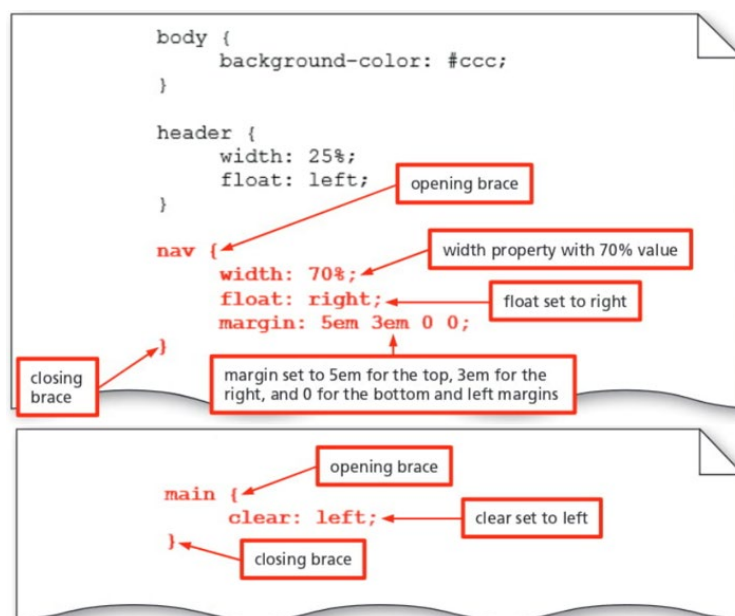
```
body {
        background-color: #ccc;
}

header {
        width: 25%;
        float: left;          opening brace
}

nav {                          width property with 70% value
        width: 70%;
        float: right;          float set to right
        margin: 5em 3em 0 0;
}                              margin set to 5em for the top, 3em for the
closing                        right, and 0 for the bottom and left margins
brace
```

```
                               opening brace
main {
        clear: left;           clear set to left
}                   closing brace
```

Figure 4

- Save your changes.
- Reload your **index.html** file in your browser to see the effects.

Part 5 – Coding the Navigation Links as an Unordered List

Coding links in a navigation system as an unordered list is a common way to semantically describe the content because a list of links is truly an unordered list. This technique also provides the structure, or "box," around each link, which is required to style the links in different ways for different viewport sizes later. In this part, you will code the navigation links on the **Home, About Us, Contact Us,** and **template pages** as list items in an unordered list.

- Your Turn - try to complete the following steps yourself before looking at Figure 5:

  o Open the **index.html** file in your text editor.
  o In the <nav> section, replace the paragraph opening <p> and closing </p> with unordered list opening <ul> and closing tags.
  o To code the navigation links as list items in an unordered list, add the opening <li> and closing tags to each anchor element.

Note: you can do this in VS Code by selecting the content and running the "Wrap element" command"

o Remove the nonbreaking space characters, , and vertical lines, ┃, after each closing tag.
o If you have done this correctly, your code for <nav> area should look like Figure 5.
o Copy the unordered list code in the <nav> section you have just created, starting with the opening <ul> tag, and continuing through the ending tag.
o Paste it in each of the remaining Fitness webpages.
o Save and view all four HTML files in a browser to verify the list items appear as a bulleted list.

Figure 5

Part 6 – Styling the Navigation Using CSS

The default marker for each list item in an unordered list is a bullet. You will add style rules to remove the bullet from the unordered list and to display the list items as an inline block so that they appear horizontally on the page. You will do this because, for a desktop display, you want the links in the navigation bar to appear on a single line, and you do not want to display the bullet marker.

- In your **styles.css file**, add the following code (Figure 6):

```
nav ul { /* add the ul selector nested within a nav element */

        list-style-type: none;  /* Remove the bullet marker */
        margin: 0; /* Remove the margin */
        text-align: right; /* Align the unordered list to the right */
}
```
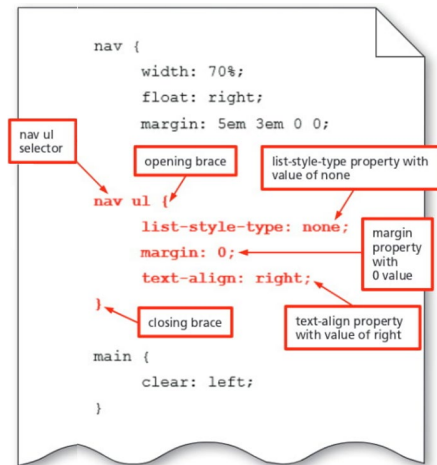
Figure 6

- Your Turn:
  - Add a style rule for the **nav li** selector:
    - Set to **display** (property) as **inline-block** (value)
    - Set font size to 1.75em

  - Add a style rule for the **nav li a** selector:
    - Set to display as block
    - Set colour to #fff
    - Set text align to centre
    - Set top and bottom paddings to 0.5em, and right and left paddings to 1em
    - Set text decoration to none

  - Save the **styles.css** file and view the **index.html** page in your browser to see the effects.

Part 7 – Adding a Hero Image

You want to capture the attention of your audience by including visual content to clarify the purpose of the website. For this part, you will add a hero image to the home page of the Forward Fitness Club website.

- Below the </nav> tag, add the following code (Figure 7):

```
</nav>

<!-- Hero Image -->
<div id="hero">
        <img src="images/hero-image.jpg" alt="left arm extended holding a weight">
</div>

<main>
```

```
</nav>

<!-- Hero Image -->
<div id="hero">
    <img src="images/hero-image.jpg" alt="left arm extended holding a weight">
</div>

<main>
```

Figure 7

- Remove the height and width attributes and values from the logo image as can be seen in Figure 8.
  - You remove the height and width attributes and use CSS to create a responsive image that grows and shrinks with the size of the browser window.

Figure 8

- Add a style rule for the **img** selector using the following code (Figure 9):

```
img {
        max-width: 100%
        display: block;
}
```

- Save changes to all files and then open **index.html** in your browser to view the page.

Figure 9

Part 8 – Removing Height and Width Attributes from img Elements

You will remove the height and width attributes and use CSS to create a responsive image that grows and shrinks with the size of the browser window.

- Open the **about.html**, **contact.html**, and **template.html** files in your text editor, delete the height and width attributes and values from the image elements.

Part 9 – Modifying the Style Rule for the Main Element

You already have a style rule for the main element, but need to modify it to complete its design. Currently, the style rule for the main selector only has one declaration. Add more declarations to enhance the appearance of the main content on the webpage.

- Within the main CSS selector, add the following code (Figure 10):

  background-color: #fff;
  padding: 2%;
  font-size: 1.25em;

- Save your changes and view the home page in a browser to see the applied styles. If necessary, scroll down to view the content within the main element.
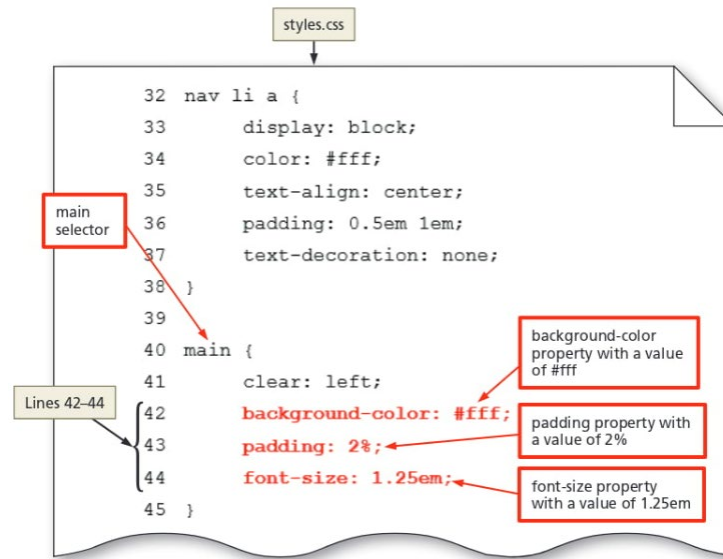
Figure 10

Part 10 - Creating a Style Rule for the Footer Element

The footer is located at the bottom of the webpage below the main area and contains paragraph elements with copyright information and an email address. Footers usually provide supplemental information, so text could be smaller than the other text on the page. Our footer should also be centred and have some padding (active white space)to improve its appearance and legibility. You will create two style rules to style the footer paragraph content and the email link.

- In **styles.css**, under the closing brace for the main selector, add the following code:

```css
footer p {
        font-size: 0.75em;
        text-align: center;
        color: #fff;
        padding: 0 1em;
}

footer p a {
        color: #fff; /* Sets the link colour */
        text-decoration: none;
}
```

- Save your changes and view the home page (**index.html**) in a browser.

Part 11 - Creating Style Rules for IDs on the About Us Page

The content on the About Us page is displayed in a single column. Improve the layout of this page by dividing the page content into three columns. Improving the layout of the webpage will improve the overall appearance and show more content within the same horizontal space.

- In the **styles.css** file, under the closing brace for the main selector, add the following code (Figure 11):

```
#weights, #cardio, #training {
        width: 29%;
        float: left;
        margin: 0 2%;
}
```
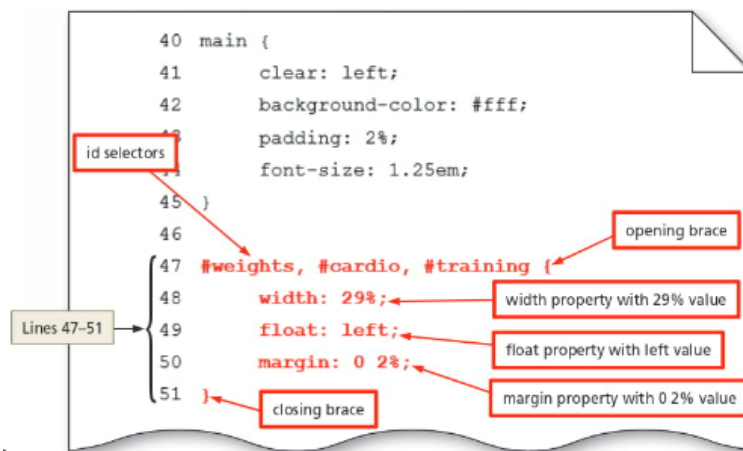

Figure 11

- Add the style rule for the **#exercises** id selector using the code below (Figure 12).

```
#exercises {
        clear: left;
        border-top: 1px solid #000;
        border-bottom: 1px solid #000;
        background-color: #f2f2f2;
        padding: 1% 2%;
}
```
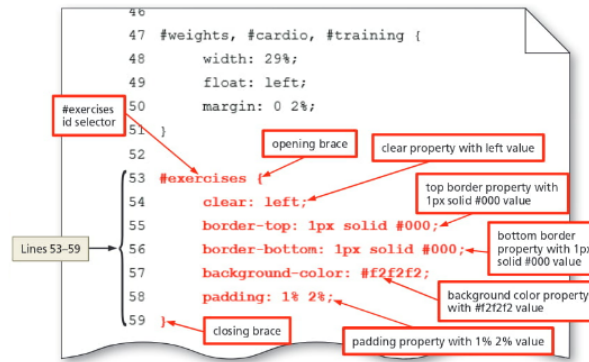
Figure 12

- Add the style rules for **#exercises dt** and **#exercises dd** selectors using the following code (Figure 13).

```
#exercises dt {
        font-weight: bold;
}

#exercises dd {
        padding:  0.5% 1% 2% 0;
}
```
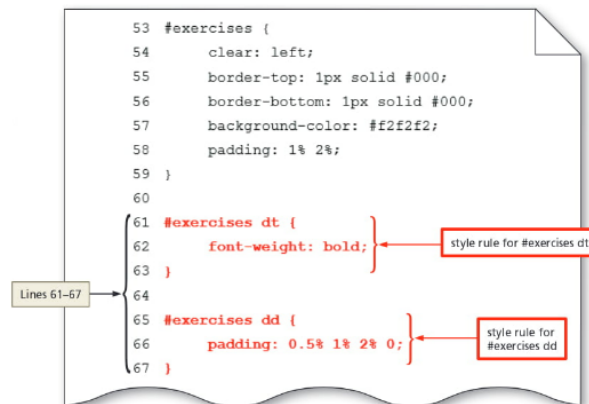

Figure 13

Part 12 - Creating Style Rules for IDs on the Contact Page

The content on the Contact Us page is aligned left. You want to change the text alignment to centre the content within the main element. This is because centring the content will improve the overall page appearance.

Create two new style rules:
- Create a style rule for the contact id attribute to centre-align its content.
- Create a style rule for the email link within the contact id attribute to change its colour and remove the underline.

Part 13 - Creating and Styling the Action Class

The home page includes the text, "start your free trial!" Enhance this as a call-to-action item by wrapping it in a span element and then assigning a class attribute to the span element to style the text. Styling the call-to-action text differently from the rest of the paragraph will attract the audience's attention. The following steps create a span element and a style rule for a class selector.

- Switch to **index.html** and place your insertion point before the word "**start**" (Figure 14) and insert the following code to insert an opening span tag with a class attribute value of action:

  `<span class="action">`

- Place your insertion point to the right of the exclamation point after the word trial, and then type `</span>` to insert a closing span tag (Figure 14).

```
<p>We provide a FREE one-week membership so you can experience
   the benefits of our  equipment and facility. This one-week
   trial gives you complete access to our  equipment, training
   classes, and nutrition planning. Contact us today to
   <span class="action">start your free trial!</span></p>
</div>

</main>
```

Figure 14

- Switch to **styles.css**, add the following code (Figure 15) and save the file.

```
.action {
        font-size: 1.35em;
        color: #666600;
        font-weight: bold;
}
```
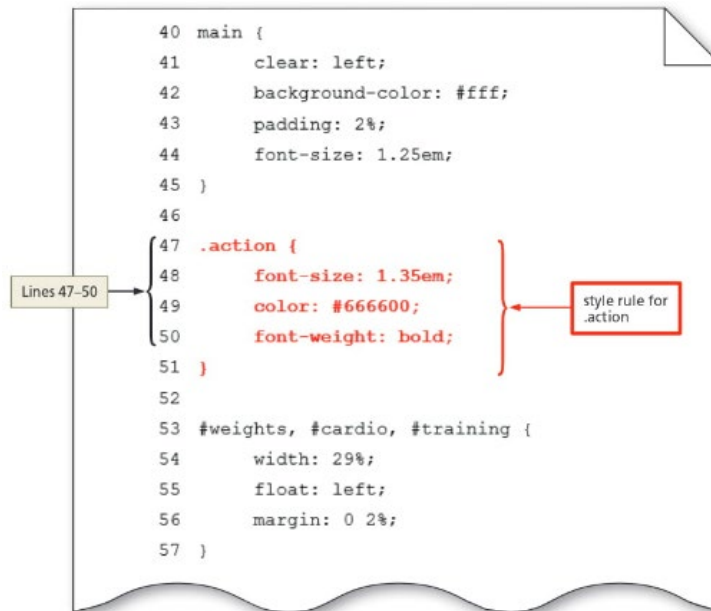
Figure 15

- Save all changes and view **index.html** in your browser.

Part 14 - Creating a CSS Reset and Improving the Appearance of the Forward Fitness Club Website

Every browser uses its own default style sheet to display a webpage. Unfortunately, all browsers do not use the same style rules within their style sheets. This can cause the appearance of a webpage to vary slightly when you view the page in different browsers. To help resolve webpage display inconsistencies, you will create a style rule known as a CSS reset.

- Create a CSS reset style rule that specifies a zero value for margin, padding, and border.
- Switch to the **styles.css** file, under the comment add the following code (Figure 16):

```
/*
Author:  followed by your first and last name
Date:    and include the current date
File Name: styles.css
*/

/* CSS reset */
body, header, nav, main, footer, h1, div,  img, ul {
        margin: 0;
        padding: 0;
        border: 0;
}
```

```
body, header, nav, main, footer, h1, div, img {
        margin: 0;
        padding: 0;
        border: 0;

}


body {
        background-color: #ccc;

}
```
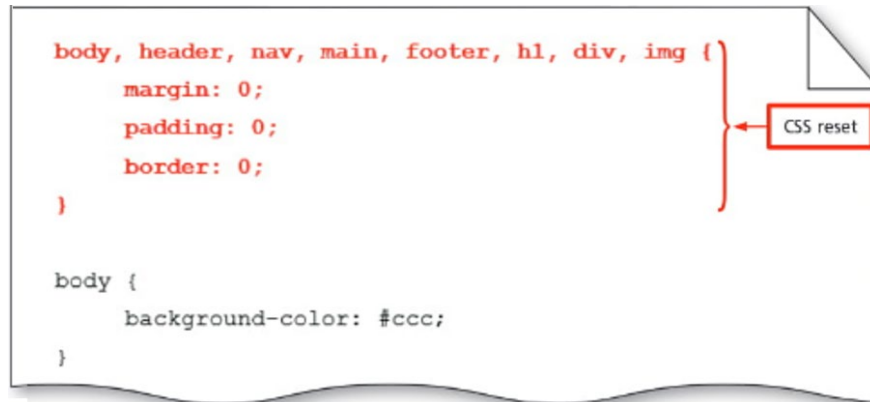
CSS reset

Figure 16

- Open **index.html** in your text editor.
- Delete the heading 1 element (**<h1>Forward Fitness Club</h1>**). See Figure 17.
    - Repeat the step with **about.html**, **contact.html**, and **template.html** (Figure 17).
- Save all changes and view the HTML files on your browser.

```
<div id="container">
    <!-- Use the header area for the website name or logo -->
    <header>
        <a href="index.html"><img src="images/forward-fitness-logo.png" alt="Forward Fitness Club logo"></a>
    </header>
```

Figure 17

- Change the **body** selector background colour to black.

- For the selector **nav li**:
    - Change font size to 1.5em
    - add the font family with value: Verdana, Arial, sans-serif
    - add the font weight with value: bold

- Add the style rules for #contact:
    - Set text align to centre

- Add the style rules for #contact a:
    - Set colour to #666600
    - Set text decoration to none

Part 15 – Improve the Appearance of Your Training Zone Webpages by Using CSS

- **Your Turn** - Now that you have learnt and applied CSS to the Fitness webpages, you can try applying CSS to your Training Zone webpages to improve their appearance. Don't forget to refer to week 1-3 pracs.

Deliverables:

- **A single zip file** containing
    - o **The Fitness folder**. The "Fitness" folder includes:
        - o HTML files (index.html, template.html, contact.html, and aboutus.html) you have created for the site
        - o a CSS file for the "Fitness" website
        - o the "images" folder contains images for the "Fitness" website
    - o **The Training Zone folder.** The "Training Zone" folder includes:
        - o HTML files you have created for the site
        - o a CSS file for the "Training Zone" website
        - o the "images" folder contains images for the "Training Zone" website